

(19) 日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11) 特許出願公開番号
特開2000－82004
(P2000－82004A)

(43) 公開日 平成12年 3 月21日 (2000. 3. 21)

(51) Int.Cl.⁷
G 0 6 F 12/00

識別記号
5 3 1

F I
G 0 6 F 12/00

テーマコード (参考)
5 3 1 D
5 3 1 M

審査請求 未請求 請求項の数12 O L (全 22 頁)

(21) 出願番号	特願平11－160557	(71) 出願人	000005108 株式会社日立製作所 東京都千代田区神田駿河台四丁目 6 番地
(22) 出願日	平成11年 6 月 8 日 (1999. 6. 8)	(72) 発明者	菊池 俊彦 神奈川県横浜市戸塚区戸塚町5030番地 株 式会社日立製作所ソフトウェア事業部内
(31) 優先権主張番号	特願平10－193127	(72) 発明者	三木 隆司 神奈川県横浜市戸塚区戸塚町5030番地 株 式会社日立製作所ソフトウェア事業部内
(32) 優先日	平成10年 7 月 8 日 (1998. 7. 8)	(74) 代理人	100075096 弁理士 作田 康夫
(33) 優先権主張国	日本 (J P)		

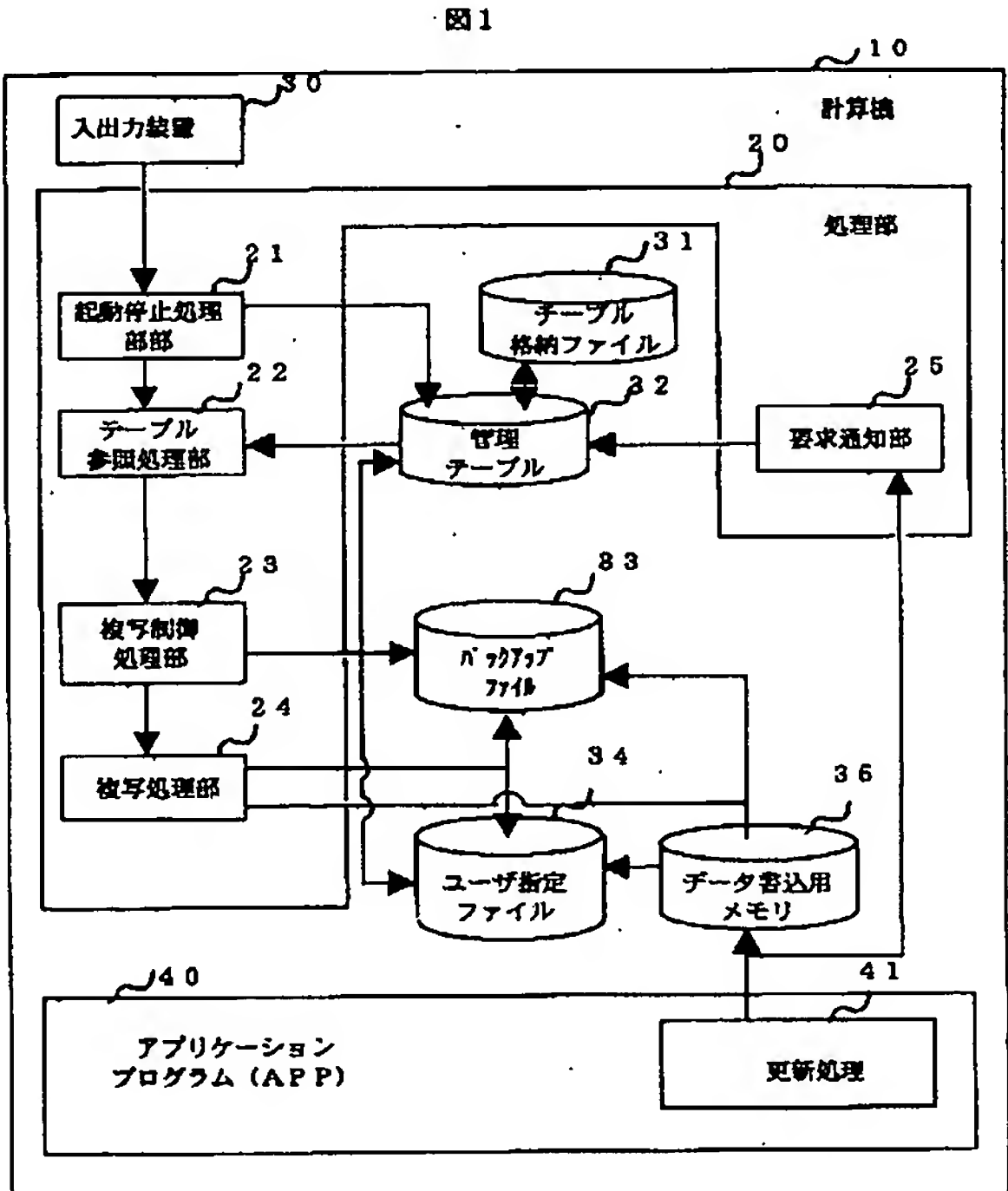
最終頁に続く

(54) 【発明の名称】 データ 2 重化システム

(57) 【要約】

【課題】従来の技術では、ファイルのバックアップと回復を、ユーザニーズに合わせて、複数のファイルのそれぞれに固有の条件に基づき行なうことやアプリケーションから任意に行なうことができない。

【解決手段】ユーザが指定したファイルについてのみソフトウェアにより複数の手順でバックアップおよび回復を行うシステムであって、操作の対象ファイルの名称とバックアップ手順および回復手順を対応付けて管理テーブル 3 2 に常駐させ、対応付けられた手順に応じて、複写制御処理 2 3 により、指定ファイルのバックアップと回復の契機となる事象の発生や、要求通知部 2 5 を経由した A P P 4 0 からの管理テーブル 3 2 に対するバックアップや回復実行の要求発生を監視し、事象や要求が発生した場合、複写処理 2 4 を起動してバックアップもしくは回復のための複写を行う構成とする。



【特許請求の範囲】

【請求項１】 計算機で使用するデータをバックアップするデータ２重化システムであって、
前記計算機で稼動するアプリケーションプログラムのデータ更新を監視するデータ更新監視手段と、前記データ更新監視手段が監視する対象を定義する監視条件定義手段と、前記アプリケーションプログラムがアクセスする記憶手段と、前記データ更新内容を前記記憶手段中に複写する複写手段とを備えることを特徴とするデータ２重化システム。

【請求項２】 請求項１のデータ２重化システムにおいて、前記監視する対象に入出力エラーが発生したときに前記複写手段で複写した前記データ更新内容を前記監視する対象に反映しデータを回復する回復手段を備えることを特徴とするデータ２重化システム。

【請求項３】 請求項１のデータ２重化システムにおいて、前記監視条件定義手段は前記データ更新内容を複写するタイミングを定義できることを特徴とするデータ２重化システム。

【請求項４】 請求項３のデータ２重化システムにおいて、前記複写するタイミングは前記データが更新されたときと前記データを含むファイルが更新後にクローズされたときの少なくともいずれかのタイミングを含むことを特徴とするデータ２重化システム。

【請求項５】 請求項２のデータ２重化システムにおいて、前記回復手段がデータの回復を行う条件を定義する回復条件定義手段を備えることを特徴とするデータ２重化システム。

【請求項６】 請求項５のデータ２重化システムにおいて、前記回復条件定義手段はデータの入出力障害が発生したとき、前記データをアクセスするアプリケーションプログラムからのデータ回復指示があったとき、ユーザからのデータ回復指示があったときのうち少なくとも一つの条件を定義できることを特徴とするデータ２重化システム。

【請求項７】 計算機で使用するデータをバックアップするデータ２重化方法であって、
データ更新監視手段は前記計算機で稼動するアプリケーションプログラムのデータ更新を監視するステップと、データの更新があったとき複写手段が前記アプリケーションプログラムのアクセスする記憶手段中に前記データの更新内容を複写するするステップとを含むことを特徴とするデータ２重化方法。

【請求項８】 請求項７のデータ２重化方法において、前記データ更新監視手段が監視する対象の指定を受け付けるステップを含むことを特徴とするデータ２重化方法。

【請求項９】 請求項８のデータ２重化方法において、回復手段が前記監視する対象に入出力エラーが発生したときに前記複写手段で複写した前記データ更新内容を前記記憶手段中に複写するステップとを備えることを特徴とするデータ２重化方法。

ることを特徴とするデータ２重化方法。

【請求項１０】 請求項７に記載の方法を実現するプログラムを格納する計算機に読み取り可能な記憶媒体。

【請求項１１】 請求項８に記載の方法を実現するプログラムを格納する計算機に読み取り可能な記憶媒体。

【請求項１２】 請求項９に記載の方法を実現するプログラムを格納する計算機に読み取り可能な記憶媒体。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】 本発明は、コンピュータシステムにおけるファイルの２重化（バックアップと回復）技術に係わり、特に、計算機のファイルのバックアップに好適なファイル２重化システムに関するものである。

【０００２】

【従来の技術】 従来のファイルのバックアップ技術では、ハードディスクドライブ（ＨＤＤ）の２重化などにより、全体ファイルのバックアップを常時取得している。そのため、ユーザのハードウェア設備投資が必要であり、バックアップのコストが高くなってしまう。

【０００３】 その結果、このバックアップ技術は、大型計算機やワークステーションへの適用が主であり、近年低価格化したパーソナルコンピュータへの適用は費用の面からコスト高となる問題があった。

【０００４】 このような問題を解決する技術として、例えば、特開平９－６５４５号公報（「ユーザファイルバックアップシステム」）に記載のものがある。この技術では、ハードディスク内の任意のファイルに対して事前に指定したエラーの発生を監視し、エラー検知後のファイルアクセスリトライ回数が事前に指定したリトライ上限回数に達した時点で、当該ファイルのバックアップを行う技術がある。この技術によれば、ファイルアクセスリトライ回数が事前に指定したリトライ上限回数に達したファイルのみに限定してバックアップ対象とするので、少ないハードディスク容量でのファイル２重化を実現できる。

【０００５】

【発明が解決しようとする課題】 上記従来技術では以下の点が考慮されていない。

【０００６】 従来技術では、エラーのリトライ回数が、事前に指定されたエラーのリトライ上限回数（指定値）を超過した時点で初めてバックアップを取得するので、エラーのリトライ上限回数を超過する前にファイルの情報を回復する必要が生じた場合にはバックアップファイルが存在しないか存在した場合でも必ずしも最新のバックアップではないことになる。

【０００７】 つまり、ファイルのバックアップを行う場合、バックアップ対象となるファイルの内容や用途によってはバックアップ情報に要求される新しさが異なる。これに対応するには、例えば、バックアップ対象ファイルの更新履歴を管理し、バックアップする際に最新のバージョンのみをバックアップする。

プを取得するような高い頻度でのバックアップ取得や、ファイルへのデータ書き込みが終了し、ファイルクローズが行われた場合にバックアップを取得するような、前者より低い頻度でのバックアップ取得など、複数の異なる頻度でバックアップを取得する必要がある。

【０００８】また、バックアップ対象ファイルが複数の場合は、各ファイルと複数のバックアップ取得条件とを対応付けて管理することが必要となる。

【０００９】さらにバックアップ対象のファイル数が多い場合には回復に時間がかかり、データの更新が頻発するような用途で用いられるファイルについて手動によるデータ回復を行うことは困難である。

【００１０】また、バックアップ取得やファイル回復の実現が、計算機内のアプリケーションプログラム（以下ＡＰＰと記載する）への改造が必要な方式による場合には、ユーザのプログラム開発や保守能力によってはバックアップのためのＡＰＰの改造が難しい場合がある。

【００１１】本発明の目的は、ファイルのバックアップと回復を、ユーザニーズに合わせて、複数のファイルのそれぞれに固有の条件に基づきＡＰＰから独立に行なうことである。

【００１２】また本発明の他の目的は、少ないハードディスク容量でのファイル２重化すなわち安価なハードウェアの設備投資でのファイル２重化を、ユーザニーズに応じて効率的かつ高信頼に行なうことが可能なファイル２重化システムを提供することである。

【００１３】

【課題を解決するための手段】上記目的を達成するため、本発明のファイル２重化システムは、ユーザが指定した任意のファイル（指定ファイル）についてのみ、ＡＰＰからのその指定ファイルに対するデータ入出力およびその際のエラーの発生を監視し、監視の対象となる事象（データ入出力とその際のエラー）が発生した場合には、指定ファイルのバックアップファイルへの複写またはバックアップファイルからユーザ指定ファイルへの複写をＡＰＰとは独立のソフトウェアにより行い、指定ファイルのバックアップおよび回復を行なう。

【００１４】特に本発明では、バックアップおよび回復について各々複数の実行手順を持ち、複数の指定ファイルのそれぞれに対して個別に実行されるバックアップ処理手順と回復処理手順を、各々バックアップ区分と回復区分などとして定義し、各区分と指定ファイルとの対応付けをコンピュータのメモリ上にテーブル形式で格納する。これにより、ファイル２重化を、ユーザニーズに応じて効率的に行なうことができる。

【００１５】このバックアップ区分としては、指定ファイル毎に起動するこの指定ファイルの更新後のクローズを監視する処理により、指定ファイルを更新してクローズしたときにバックアップを取得する手順と、指定ファイルへのデータ書き込みが終了し、ファイルクローズが行われた場合にバックアップを取得するよう

より指定ファイルにデータ書き込みが発生したときバックアップを取得する手順の２つを有する。

【００１６】また、ファイル回復区分としては、指定ファイルにデータ入出力の障害が発生した時点で、指定ファイル毎に起動する、指定ファイルのデータ入出力時の障害発生を検知することにより、指定ファイルにデータ入出力の障害が発生した時点で、ファイルを回復する手順と、任意の契機でユーザの手動操作によりファイルを回復する手順と、指定ファイルを使用する任意のＡＰＰから要求されたときにファイルを回復する手順の３つを有する。

【００１７】

【発明の実施の形態】以下、本発明の実施例を、説明する。

【００１８】図１は、本発明のファイル２重化システムの一実施例を示すブロック図である。

【００１９】本実施例のファイル２重化システムは、パーソナルコンピュータ等の計算機１０内に構築され、この計算機１０は、ＣＰＵ（Central Processing Unit）により蓄積プログラム方式でのファイルの２重化等の処理を行う処理部２０、ディスプレイおよびキーボード等からなる入出力装置３０、ハードディスクドライブ（ＨＤＤ）内のテーブル格納ファイル３１とバックアップファイル３３および２重化の対象となるユーザ指定ファイル３４、主メモリ上の管理テーブル３２とデータ書込用メモリ３５、ユーザ指定ファイル３４にデータを読み書きするための更新処理４１を有するＡＰＰ４０とにより構成されている。

【００２０】処理部２０は、処理部２０の起動と停止を制御する起動停止処理部２１、管理テーブル３２に記録されている内容を参照するテーブル参照処理部２２、管理テーブル３２の内容に応じてユーザ指定ファイル３４に対するデータ更新とエラー発生とＡＰＰ４０からの要求通知の発生を監視して２重化の実行条件に適合した場合に複写処理を起動する複写制御処理部２３、この複写制御処理部２３から起動され実際にファイルの２重化を実行する複写処理部２４、ＡＰＰ４０からのファイル２重化要求を受け付ける要求通知部２５から構成されている。

【００２１】このような構成により、本例のファイル２重化システム（以下「本システム」と記載）は、次のようにしてファイルの２重化を行う。

【００２２】まず、計算機１０の起動時または入出力装置３０からのユーザ要求時に、本システムの起動停止処理部２１が、ファイルを２重化するための情報として、ユーザ指定ファイル名とバックアップファイル名、バックアップ区分と回復区分、およびその他の情報のユーザ入力の受け付けと、テーブル格納ファイル３１から管理テーブル３２へのユーザ入力済み情報の展開、および、

【0023】起動されたテーブル参照処理部22は、管理テーブル32に記録されているユーザ入力情報を参照し、複写制御処理部23を起動して、この複写制御処理部23に、参照したテーブル内容を通知する。

【0024】複写制御処理部23は、通知されたテーブル内容から、バックアップ区分と回復区分に応じて、ユーザ指定ファイル34の更新の発生およびエラー発生と、APP40からのファイル2重化要求の発生を監視し、バックアップおよび回復実行の条件に合致した場合は、複写処理部24を起動し、この複写処理部24に、2重化実行のための情報としてテーブル内容を通知する。

【0025】起動された複写処理部24は、複写制御処理部23から通知された情報を元に、ユーザ指定ファイル34とバックアップファイル33間のファイル複写、およびデータ書込用メモリ35からバックアップファイル33へのデータ複写を実行する。

【0026】また、APP40からの要求によりバックアップおよび回復を行う場合は、更新処理41から要求通知部25を呼び出し、ユーザ指定ファイル名およびユーザ指定ファイルに対する更新データの先頭アドレスとデータ長を渡す。要求通知部25は、管理テーブル32に、渡された要求の発生と更新データ先頭アドレスおよびデータ長を記録する。

【0027】次に、このようなファイル2重化システムの各構成要素について、その内容と構造および処理の詳細を説明する。

【0028】まず、図2を用いて、本システムにより実施されるバックアップ区分と回復区分について説明する。

【0029】図2は、図1における管理テーブル内のバックアップ区分および回復区分の構成を示す説明図である。

【0030】本例は、バックアップ区分および回復区分に対する本システムの動作の対応付けを示している。

【0031】すなわち、バックアップ区分としては、「更新終了」と「更新中」の2つを有し、「更新終了」が指定された場合は、図1のユーザ指定ファイル34が更新後にクローズされた時点でバックアップを行い、また、「更新中」が指定された場合はユーザ指定ファイル34の更新可能権限でオープン中にデータ書込みが発生した時点でバックアップを実行する。

【0032】また、回復区分としては、「自動」と「手動」と「通知」の3種類を有し、「自動」が指定された場合は、ユーザ指定ファイル34に対するデータ入出力時にエラーが発生した時点で回復を実行する。また、

「手動」が指定された場合は、ユーザが手動操作でバックアップファイル34から回復を行い、さらに、「通知」が指定された場合は、ユーザ指定ファイル34を利用するときにエラーが発生した時点で回復を実行する。

【0033】次に、更新終了指定時と更新中指定時のバックアップ動作の違いを、図3により説明する。

【0034】図3は、バックアップ区分毎の複写動作の違いを示す図である。

【0035】特に、更新終了指定時と更新中指定時のそれぞれにおけるバックアップ動作の違いを示すものである。

【0036】更新終了指定時は、ユーザ指定ファイル34の更新クローズが発生した時点（a）で、ユーザ指定ファイル34全体をバックアップファイル33に上書き複写することでバックアップを行う（b）。

【0037】これに対し、更新中指定時は、ユーザ指定ファイル34の更新中にデータ書込みが1件発生する都度に（c）、同一のデータをバックアップファイル33に追加複写することでバックアップを行う（d）。

【0038】次に、図1における処理部20の起動および停止を制御し、ユーザ入力を受け付ける起動停止処理部21について図4と図5を用いて説明する。

【0039】図4は、図1における起動停止処理部の主処理動作例を示すフローチャートであり、図5は、図1における起動停止処理部による入力処理動作例を示すフローチャートである。

【0040】本例での処理は、図1の計算機10の起動時、または入出力装置30からのユーザ要求により、処理部20中で最初に起動され、主処理と、ユーザ入力を受け付ける入力処理とからなる。主処理については図4で、また、入力処理については図5で説明する。

【0041】主処理はユーザから入力される情報を格納するための管理テーブルをメモリ中にローディングし、そのテーブルの内容を参照するテーブル参照処理を起動するための処理である。

【0042】図4において、主処理の起動後、まず管理テーブルをメモリ中にローディングする（ステップS41）。次にユーザ入力を受け付けるためのオプション入力処理を起動する（ステップS42）。その後、上記テーブル内を参照するためにテーブル参照処理を起動し（ステップS43）、主処理を終了する。次に、図5を用いて、ユーザからの入力情報の内容をチェックし、図1の管理テーブル32への書込みを行う入力処理について説明する。尚、ユーザ入力情報の内容については、図6に示す。また、この図6中のユーザが直接入力しない情報については後述の各処理中で説明する。

【0043】図5において、本処理の起動後、バックアップ等のオプションを設定するためのユーザ入力要求があるか否かを判別し（ステップS51）、要求がなければ終了し、要求があれば入出力装置30上に入力画面を表示する（ステップS52）。

【0044】この入力画面では、ユーザが直接入力する情報について、図6に示すように、バックアップ区分、回復区分、通知区分の3つを指定する。

復区分603、ユーザ指定ファイル名606、バックアップファイル名607、および、再実行回数610などの入力を受け付ける。

【0045】図6に示す再実行回数610は、バックアップ処理および回復処理が正常終了しなかった場合に、最大何回まで再実行するかを指定するためのものである。また、バックアップ区分602と回復区分603およびユーザ指定ファイル名606については必須入力とし、さらに、バックアップファイル名607と再実行回数610についてはユーザ入力を省略可能とし、その際は、本システムがファイル名称と回数を仮定する。

【0046】図5においては、その後、入力されたバックアップ等の条件を示す情報が正しいかどうかをチェックし（ステップS53）、誤りがあればステップS52の入力画面表示に戻りユーザの再入力を促し、正しければ入力された情報を管理テーブル32に書き込む（ステップS54）。

【0047】次に、ユーザ入力が行われたことにより、ユーザ指定ファイル名の新規登録か更新か削除かのいずれかが行われテーブル内容に変更が生じているものとし、最新のテーブル内容を参照するため図1のテーブル参照処理部22を起動する（ステップS55）。

【0048】尚、既に登録されている内容の更新の場合は、バックアップファイル名とリトライ回数のみをその更新対象として許可し、ユーザ指定ファイル名の変更は新規登録として扱う。

【0049】このようなバックアップ対象ファイルを示したユーザ指定ファイル34と、それに対して行われるバックアップ区分および回復区分との対応付けを管理する管理テーブル32の内容を、図6を用いて説明する。

【0050】図6は、本実施例における管理テーブルの構成例を示す。

【0051】管理テーブル32は、2重化を行う対象と2重化のオプションを対応付けた情報を1エントリとして保持する構成とする。

【0052】各項目毎の機能について、以下、順次説明する。

【0053】まず、横軸は、「プロセス番号」601、「バックアップ区分」602、「回復区分」603、「処理種別」604、「処理開始時間」605、「ユーザ指定ファイル名」606、「バックアップファイル名」607、「実行結果フラグ」608、「処理中フラグ」609、「再実行回数」610、「エントリ更新種別」611、「要求通知有無」612、そして、「更新情報格納キュー領域」614の各項目からなる。

【0054】ここで、「プロセス番号」601は、実際に複写を行う図1の複写処理部24のプロセスを特定するために用いられ、また、「処理種別」604は、複写処理部24が正常終了しなかった場合にバックアップと回復のどちらが行われるかを判別するために用い

れる。また、「処理開始時間」605は、複写処理部24が開始された時間をもとに処理に不当に長い時間を要しないか監視するために用いられる。

【0055】また、「実行結果フラグ」608は、複写処理部24が正常に終了したか否かを判別するために用いられ、「処理中フラグ」は、ユーザ指定ファイル34に対しての更新発生およびエラー発生の監視状態にあるか否かを判別するために用いられる。また、「再実行回数」610は、複写処理24が失敗した場合に最大で何回再実行を行うか規定するために用いられる。

【0056】さらに、「エントリ更新種別」611は、登録済みのエントリがユーザ入力により変更される場合に、その変更の形態を示すために用いられ、「要求通知有無」612は、APPからの要求が通知されたか否かを示すために用いられる。

【0057】そして、管理テーブル32の最後の項目としての「更新情報格納キュー領域」613は、「バックアップ区分」602に更新中が指定された場合に、ユーザ指定ファイル34に書き込まれるデータを1件毎にバックアップするために、データ1件毎のデータ書き込み用メモリ35上の位置情報とユーザ指定ファイル34の更新後のクローズの発生と、その結果を格納するキュー領域の先頭アドレスを格納するために用いられる。

【0058】この「更新情報格納キュー領域」613は、ユーザが指定したファイルに書き込まれるデータを一件ごとに取得するために、メモリ中のデータの先頭アドレスとデータ長の2種類の情報を保持する。データ書き込み後に行われるユーザ指定ファイル34の更新後のクローズの発生有無とその結果を、ファイルクローズフラグ617とクローズ成否フラグ618の形式との2種類のデータ書き込みおよびファイル更新後のクローズが発生した順に、先入れ先出しのキュー形式で格納する。尚、このキュー領域は、バックアップ区分が更新中と指定されたエントリの1件毎に予め生成され、1エントリの最終項目である更新情報格納領域先頭アドレスにキュー領域の先頭アドレスを格納することで関連付けられる。

【0059】次に、このような構成の管理テーブル32の内容を参照する図1のテーブル参照処理部22について、図7を用いて説明する。

【0060】図7は、図1におけるテーブル参照処理部による処理動作例を示すフローチャートである。

【0061】本実施例の処理を行なう図1のテーブル参照処理部22の起動は、本システムの起動時とユーザ入力の発生時に行われる。

【0062】まず、本処理起動後、図1の管理テーブル32の先頭行を参照し（ステップS71）、参照したエントリのファイル監視中フラグが監視中か否かを判別する（ステップS72）。

【0063】監視中であれば、エントリ更新種別が削除

ーザから2重化解除の指示として当該エントリの削除要求が発生しているものと判断し、当該エントリを管理テーブル32上から削除し（ステップS74）、ステップS7Dの最終エントリの判別に移る。

【0064】また、ステップS73において、エントリ更新種別が削除でない場合は、更新または一時停止のいずれかであるか否かを判別し（ステップS75）、そのいずれでもない場合は、現在ユーザ指定ファイル34を監視中のエントリに対しての変更が発生していないものと判断してステップS7Dの最終エントリの判別に移る。エントリ更新種別が更新か一時停止のいずれかであった場合は、当該エントリのプロセス番号をもとに、複写制御処理の対応するサブプロセスを停止する（ステップS76）。

【0065】その後、エントリ更新種別が更新であるか否かを判別し（ステップS77）、エントリ更新種別が更新の場合はステップS7Bの複写制御処理起動に移る。エントリ更新種別が更新でない場合は、ユーザ要求がエントリの一時停止であると判断し、当該エントリの監視中フラグを未監視にしてステップS7Dの最終エントリの判別に移る。

【0066】また、ステップS72の判別で、参照したエントリが監視中でなかった場合は、未監視のエントリに対する削除が要求されたか否かを判別し（ステップS79）、エントリ更新種別が削除の場合は、当該エントリを削除しステップS7Dの最終エントリの判別に移る。

【0067】また、エントリ更新種別が削除でなかった場合は、エントリが新規に登録されたものであると判断して、ユーザ指定ファイルの監視を開始するため複写制御処理部23を起動し（ステップS7B）、管理テーブル32に指定されたユーザ指定ファイル名とバックアップファイル名とバックアップ区分と回復区分を起動した複写制御処理部23に通知する（ステップS7C）。

【0068】その後、参照したエントリが、管理テーブル32上での最終エントリか否かを判別し（ステップS7D）、最終エントリでなければ次のエントリを参照してステップS86の回復区分が手動かの判定に移る。最終エントリであった場合は、起動停止処理部21の入力処理が未起動であるか否かを判別し（ステップS7F）、未起動である場合は、入力処理を起動し（ステップS7G）、すでに起動済みの場合は、本処理を終了する。

【0069】次に、このようにしてテーブル参照処理部22が参照した内容を受け取り、バックアップおよび回復の実行条件に合致する事象の発生を監視する図1の複写制御処理部23について説明する。

【0070】図8は、図1における複写制御処理部の処理動作例を示すフローチャートである。

テーブル参照処理部22からユーザ指定ファイル名とバックアップファイル名とバックアップ区分と回復区分の情報を受け取る（ステップS81）。

【0072】そして、受け取った情報を元に、実際のファイル複写を行う複写処理部24を起動して、ユーザ指定ファイルに対する初回のバックアップを取得し、バックアップファイル33を生成する（ステップS82）。

【0073】次に、通知されたバックアップ区分が更新終了か否かを判別する（ステップS83）。

【0074】区分が更新中であれば、要求通知部25からの管理テーブル32への更新データ情報の書込みを監視し、書込みがあった場合に複写処理部24の起動を行う書込み監視処理を起動しステップS86の回復区分が手動かの判定に移る。

【0075】また、区分が更新終了であれば、ユーザ指定ファイル34の更新終了を監視し、更新終了した場合には、複写処理部24の起動を行う更新終了監視処理を起動する（ステップS85）。次に、通知されたエントリの回復区分が手動であるかを判別し（ステップS86）、手動である場合は、回復実行のための複写処理部24を起動する必要がないため終了する。

【0076】回復区分が手動でない場合は、区分が自動であるかを判別し（ステップS87）、自動でない場合は、区分が通知であると判断して、APPからの通知の発生を監視し、通知があった場合は、複写処理部24の起動を行う通知監視処理を起動する（ステップS88）。区分が自動である場合は、ユーザ指定ファイル34のデータ入出力エラーの発生を監視し、エラーが発生した場合は、回復実行のため複写処理部24の起動を行うエラー監視処理を起動する（ステップS89）。

【0077】このように、図1の複写制御処理部23は、管理テーブル32内の1エントリに対応して1つの割合でテーブル参照処理部22内から起動され、対応するエントリのバックアップ区分と回復区分に応じて各監視処理を起動する。

【0078】次に、このようにして複写制御処理部23内から起動され、実際のバックアップおよび回復のための複写を行う図1の複写処理部24について説明する。

【0079】図9は、図1における複写処理部による処理動作例を示すフローチャートである。

【0080】本処理の起動後、まず、図1の複写制御処理部23から、複写動作を決定するための情報として、複写元のファイル名または複写元データの所在、複写先ファイル名、再実行回数を受け取る（ステップS91）。

【0081】そして、複写実行時のエラーに対処するための情報として、自プロセス番号と、複写開始時刻として現在時刻を管理テーブル32上の対応するエントリの項目に書き込む（ステップS92）。

【0082】その後、図1の複写処理部24は、図1の

をもとに複写を実行し（ステップS 9 3）、正常終了したかを判別し（ステップS 9 4）、正常終了していなければ再実行回数が上限に達しているかを判別する（ステップS 9 5）。

【0 0 8 3】再実行回数が上限に達していなければ再実行回数から1回減算して（ステップS 9 6）複写を再度実行するためステップS 9 3の複写実行に移る。また、再実行回数が上限に達している場合は、管理テーブル3 2の対応するエントリの実行結果フラグに異常をセットし（ステップS 9 7）、異常がバックアップと回復のどちらで発生したかを記録するために、処理種別にバックアップまたは回復をセットし（ステップS 9 8）、ステップS 9 9のプロセス情報設定に移る。

【0 0 8 4】そして、複写実行完了まで異常に時間がかかった場合をエラーと判断する為の情報として、複写実行プロセスのプロセス番号と、複写開始時刻としてのプロセス起動時刻を処理開始時刻として管理テーブル3 2上の対応するエントリの項目に書き込む（ステップS 9 2）。

【0 0 8 5】このように、複写処理部2 4は、起動元の複写制御処理部2 3から通知される情報の内容に応じてバックアップおよび回復を実施する。

【0 0 8 6】次に、バックアップ区分が更新中である場合に、複写制御処理部2 3内から起動され、管理テーブル3 2からポイントされたキュー領域への更新データ所在の書込みを監視する書込監視処理、すなわち、図8におけるステップS 8 4の詳細について説明する。

【0 0 8 7】図1 0は、図1における複写制御処理部2 3の書込監視処理動作例を示すフローチャートである。

【0 0 8 8】本処理の起動後、まず、図6の管理テーブル3 2上で対応するエントリのキュー領域を参照する（ステップS 1 0 1）。

【0 0 8 9】そして、キュー領域への書込みの有無を判別し（ステップS 1 0 2）、書込みがない場合は、キュー領域の参照を継続し書込みの発生を監視するために、ステップS 1 0 1のキュー領域参照に移り、書込みが発生している場合は、先頭キューの内容を参照する（ステップS 1 0 3）。

【0 0 9 0】さらに、参照したキュー内容がユーザ指定ファイル3 4を更新後のファイルクローズか否かを判別し（ステップS 1 0 4）、更新後のファイルクローズでなければ複写処理部2 4を起動し参照したキュー内容を通知し（ステップS 1 0 B）、複写処理部2 4の処理時間を監視する時間監視処理を起動する（ステップS 1 0 C）。

【0 0 9 1】その後、複写処理が正常に終了したかをテーブル3 2上の実行結果フラグから判別する（ステップS 1 0 D）。

【0 0 9 2】正常終了した場合は、時間監視処理を停止し（ステップS 1 0 E）、参照した先頭キューの内容を

領域から削除し（ステップS 1 0 F）、次のキューが存在するかを判別し、存在しなければ終了する。

【0 0 9 3】また、ステップS 1 0 Iで次キューが存在すれば、この次キューを参照するためにステップS 1 0 3の先頭キュー参照に移る。

【0 0 9 4】さらに、ステップS 1 0 Dで複写処理部2 4が正常に終了しなかった場合は、再実行回数の上限に達した否かを判別し（ステップS 1 0 G）、達していない場合は、処理を継続するため新たな複写処理部2 4を起動し、現在参照のキュー内容を通知するためステップS 1 0 Bの複写処理起動に移る。リトライ回数が上限に達している場合は、データ複写が不可となる致命的な障害がファイルシステムに発生しているものとして、入出力装置3 0からエラーメッセージを出力し（ステップS 1 0 H）、本処理を終了する。

【0 0 9 5】また、ステップS 1 0 4でキュー内容がファイル更新後のファイルクローズであった場合には、ファイルクローズが成功しているか否かを当該キューのクローズ成否フラグ（6 1 8）を参照して判別し（ステップS 1 0 5）、成功の場合はステップS 1 0 Iの次キューなしかの判定に移り、失敗していれば当該エントリに指定された回復区分6 0 3が自動回復か否かを判別し（ステップS 1 0 6）、自動回復でなければステップS 1 0 Iの次キューなしかの判定に移り、自動回復の場合は複写制御処理部2 3中のステップS 8 9で起動されるエラー監視処理が起動されているか否かを判別する（ステップS 1 0 7）。

【0 0 9 6】起動していればステップS 1 0 Iの次キューなしかの判定に移り、起動していなければ、ユーザ指定ファイル3 4更新後のファイルクローズが失敗しており、ユーザ指定ファイル3 4の最後の書込みオープンからファイルクローズまでの間に書き込まれたデータを回復する必要があると判断して複写処理部2 4を起動し、複写元ファイルにバックアップファイル3 3を、複写先ファイルにユーザ指定ファイル3 4を指定して複写を実行させる（ステップS 1 0 8）。

【0 0 9 7】その後、複写処理部2 4が正常に終了したか否かを判別し（ステップS 1 0 9）、正常に終了した場合は本書き込み監視処理を終了し、失敗した場合は、ファイルシステムに致命的な障害が発生しているものとして入出力装置3 0からエラーメッセージを出力して（ステップS 1 0 A）本処理を終了する。

【0 0 9 8】次に、複写処理部2 4の所要時間を監視し、不当に長い処理時間がかかることを防止する時間監視処理、すなわち、図1 0におけるステップS 1 0 Cの処理の詳細を説明する。

【0 0 9 9】図1 1は、図1における複写処理部の実行と対に実行される時間監視処理動作例を示すフローチャートである。

【0 1 0 0】本処理は、図1の複写処理部2 4の処理に

対で起動され、複写制御処理部23の書き込み監視処理と、更新終了監視処理と、通知監視処理と、エラー監視処理中から起動される処理である。

【0101】本処理起動後、まず、起動元の処理から起動された複写処理部24のプロセス番号を受け取る（ステップS111）。

【0102】そして、受け取ったプロセス番号から複写処理部24の開始時に、管理テーブル32上に書き込まれた処理開始時刻を現在時刻から減算し（ステップS112）、事前に指定した最大処理時間を超過していないかを判別する（ステップS113）。

【0103】判別の結果、超過していなければステップS112に戻り経過時間の算出を続行し、超過していれば複写処理部24を停止し（ステップS114）、入出力装置からエラーメッセージを出力して（ステップS115）本処理を終了する。

【0104】尚、本処理は、監視対象の複写処理部24が正常または異常終了した時点で、起動元の処理から強制的に停止される。

【0105】次に、図1の管理テーブル32において、バックアップ区分に更新中が指定されていた場合の要求通知部25の処理について説明する。

【0106】図12は、図1における要求通知部の処理動作例を示すフローチャートであり、図13は、図1におけるAPPと要求通知部との間で受け渡される情報例を示す説明図である。

【0107】本要求通知部25の処理は、図1の管理テーブル32のバックアップ区分に更新中が指定されていた場合には、APP40の、ユーザ指定ファイル34の更新プロセス41中から呼び出されることにより、ユーザ指定ファイル34への書き込みデータを処理部20側に通知するものである。

【0108】実際の呼び出しの契機は、更新処理41中のユーザ指定ファイル34のオープン後に呼び出されれば良い。

【0109】まず、本要求通知部25の処理の呼び出しと同時に、更新処理41からユーザ指定ファイル名とデータ書き込み用メモリ35に書き込んだデータの先頭アドレスとデータ長、またはユーザ指定ファイル34のクローズ結果を引数として図13に示す形式で受け取る（ステップ121）。

【0110】そして、受け取った情報を管理テーブル32上の該当エントリに対応するキュー領域に書き込み（ステップS122）、書き込みが成功したか否かを判別して（ステップS123）、成功の場合は、正常値を呼び出し元の更新処理41に返却し（ステップS124）、終了する。

【0111】書き込みが失敗している場合は、再実行回数

書き込みに戻り書き込みを続行する。

【0112】また、再実行回数上限に達している場合は、呼び出し元の更新処理41にエラー値を返却し（ステップS126）、キュー領域への書き込みに際して致命的な障害が発生しているものとして、入出力装置30からエラーメッセージを出力し（ステップS127）、本処理を終了する。

【0113】尚、ステップS126で更新処理41に返却するエラー値の取り扱いについては、APP40および更新処理41の処理により任意である。また、再実行回数は、呼び出し元から引数として指定されても良く、省略時は要求通知部側で初期値を仮定する。

【0114】次に、複写制御処理部23の図6におけるステップS65で起動される更新終了監視処理について説明する。

【0115】図14は、図1における複写制御処理部で起動される更新終了監視処理動作例を示すフローチャートである。

【0116】本処理は、バックアップ区分が更新終了の場合に、ユーザ指定ファイル34の更新クローズを監視し、更新クローズ発生の際に複写処理部24を起動してバックアップを実行させるものである。

【0117】本処理の起動後、まず、複写制御処理部23から管理テーブル32上の対応するエントリの情報を受け取り（ステップS141）、ユーザ指定ファイル34に対する入出力イベントの監視を開始する。

【0118】監視中のイベントに更新後のファイルクローズの成功を検出した場合は（ステップS142）、複写処理部24を起動し（ステップS143）、受け取ったエントリ情報からユーザ指定ファイル名とバックアップファイル名を通知する。

【0119】そして、複写処理部24の時間監視処理を起動し（ステップS144）、その後複写処理部24が正常に終了したかを判別し（ステップS145）、正常に終了した場合は、時間監視処理を停止し（ステップS147）、本処理を終了する。

【0120】また、ステップS142で更新クローズの異常終了を検知した場合は、および、ステップS145で複写処理部24が正常終了しなかった場合は、ステップS146で入出力装置30からエラーメッセージを出力してステップS147の時間監視停止に移る。

【0121】次に、回復区分が通知の場合に、既存プログラムからの回復要求の発生を監視する通知監視処理について図15を用いて説明する。本処理は、図8の複写制御処理部のステップS88から起動され、要求通知部25からの通知の有無を管理テーブル32を参照することにより判別し、通知発生の場合に複写処理部24を起動する処理である。本処理の起動後、まず、複写制御処理部23から管理テーブル32上の対応するエントリ

の回復通知有無を参照し通知が発生したか否かを判別する（ステップS 1 5 2）。

【0 1 2 2】通知があった場合は複写処理部 2 4 を起動し（ステップS 1 5 3）、時間監視処理を起動し（ステップS 1 5 4）、また、通知がない場合は監視を続行する。

【0 1 2 3】その後、複写処理部 2 4 が正常に終了したか否かを判別し（ステップS 1 5 5）、正常に終了していれば時間監視処理を停止し（ステップS 1 5 7）、本処理を終了する。

【0 1 2 4】ステップS 1 5 5で複写処理部 2 4 が正常終了していなければ、入出力装置 3 0 からエラーメッセージを出力し（ステップS 1 5 6）ステップS 1 5 7の時間監視停止に移る。

【0 1 2 5】次に、回復区分が通知時の要求通知部 2 5 の処理について説明する。

【0 1 2 6】図 1 6 は、図 1 における要求通知部による処理動作例を示すフローチャートである。

【0 1 2 7】本処理は、図 1 の管理テーブル 3 2 の回復区分に通知が指定されていた場合には、APP 4 0 から呼び出されることにより、回復の実行契機を処理部 2 0 側に通知するものであり、まず、本手段の呼び出しと共に、ユーザ指定ファイル名とバックアップファイル名を引数として、呼び出し元から、図 1 3 に示す形式で受け取る（ステップS 1 6 1）。

【0 1 2 8】そして、受け取った引数の内容から管理テーブル 3 2 上の対応するエントリを求め、要求通知有無の項目に通知の発生を書き込む（ステップS 1 6 2）。その後、ステップS 1 6 2の書き込みが正常に行われたか否かを判別し（ステップS 1 6 3）、正常であれば正常値を呼び出し元に返却して（ステップS 1 6 4）処理を終了する。

【0 1 2 9】また、ステップS 1 6 3で書き込みが正常に終了しなかった場合は、再実行回数が上限に達しているか否かを判別し（ステップS 1 6 5）、達していなければ、書き込みを続行するためにステップS 1 6 2の通知領域書き込みに戻り、達している場合は、呼び出し元にエラー値を返却し（ステップS 1 6 6）、入出力装置 3 0 からエラーメッセージを出力して（ステップS 1 6 7）処理を終了する。

【0 1 3 0】尚、ステップS 1 6 4およびステップS 1 6 5で本処理で返却する正常値およびエラー値の呼び出し元での取り扱いは任意である。

【0 1 3 1】また、回復区分が通知時の本手段の呼び出し契機は 3 種類あり、APP 4 0 がユーザ指定ファイル 3 4 に対して、データ読み出しを行うか、または、データ書き込みを行うかで異なる。

【0 1 3 2】例えば、データ読み出し時は、APP 4 0 のユーザ指定ファイル 3 4 を参照する処理がエラーを検知した時点でエラーメッセージを出力する。

【0 1 3 3】またデータ書き込み時は、バックアップ区分に指定されている内容毎に契機が異なり、バックアップ区分が更新中の場合はデータ 1 件の書き込み毎にバックアップが取得されているため、呼び出し元がユーザ指定ファイルに対してのデータ書き込みがエラーになったことを検知した都度に呼び出せば良く、バックアップ区分が更新終了の場合は、呼び出し元がユーザ指定ファイルの更新クローズのエラーを検知した都度に呼び出せば良い。

【0 1 3 4】次に、複写制御処理部 2 3 の図 8 におけるステップS 8 9で起動されるエラー監視処理について説明する。

【0 1 3 5】図 1 7 は、図 1 における複写制御処理部で起動されるエラー監視処理動作例を示すフローチャートである。

【0 1 3 6】本処理は、ユーザ指定ファイル 3 4 に対するデータ入出力のエラー発生を監視し、エラーが発生した場合に、複写処理部 2 4 を起動して、回復を実行させるためのものである。尚、ここで監視するエラーは、データ読み出しエラーとデータ書き込みエラーおよびデータ書き込み後のファイルクローズのエラーである。

【0 1 3 7】本処理の起動後、まず、複写制御処理部 2 3 から対応するエントリの情報を受け取る（ステップS 1 7 1）。次に、受け取った情報から該当するユーザ指定ファイル 3 4 の監視を開始し、エラー発生の有無を判別する（ステップS 1 7 2）。エラーが発生している場合は、エントリ情報を通知して複写処理部 2 4 を起動する（ステップS 1 7 3）。

【0 1 3 8】そして、複写処理部 2 4 が正常に終了したかを判別し（ステップS 1 7 4）、正常に終了していない場合は、再実行回数の上限に達しているか否かを判別し（ステップS 1 7 5）、上限に達していない場合は、再度複写処理部 2 4 を起動するために主ルーチンへ戻る。また、上限に達している場合は、入出力装置 3 0 からエラーメッセージを出力し（ステップS 1 7 6）本処理を終了する。

【0 1 3 9】以上、図 1 ～図 1 7 を用いて説明したように、本実施例のファイル 2 重化システムでは、ユーザが指定したファイルについてのみソフトウェアでファイルをバックアップする。このことにより、ファイルシステムの 2 重化を低コストで実現することができる。

【0 1 4 0】また、バックアップ処理の手順として、ユーザ指定ファイルが更新後にクローズされた契機でバックアップを取得する手順と、ユーザ指定ファイルの更新中にデータ書き込みが発生した契機でバックアップを取得する手順の 2 つを有することにより、バックアップの対象となるファイルの内容や用途によってバックアップ情報に要求される新しさが異なる場合、例えば、バックアップ対象となるファイルへのデータ書き込みが一件発生する都度にバックアップを取得するような高い頻度でのバックアップが可能である。

了してファイルクローズが行われた都度にバックアップを取得するような前者より低い頻度でのバックアップ取得など、複数の異なる頻度でバックアップを取得する必要がある場合にも、また、バックアップ対象ファイルが複数である場合は各ファイルと複数のバックアップ取得契機との対応付けを先に述べたテーブルにより管理することができるため、ユーザの業務に応じてバックアップ運用のカスタマイズが可能となる。

【0141】さらに、回復処理の手順として、ユーザ指定ファイルに障害が発生した時点でファイルを回復する手順と、任意の時点でユーザの手動操作により回復を実行する手順と、ユーザ指定ファイルを利用する任意のプログラムから要求された契機でファイルを回復する手順の3つを有する。このように、ファイル回復のためのA P Pへの改造や新たなプログラムを作成することなく、ファイル回復を行なうことができ、ユーザの負担を少なくすることができる。

【0142】尚、本発明は、図1～図17を用いて説明した実施例に限定されるものではなく、その要旨を逸脱しない範囲において種々変更可能である。例えば、本例では、図1に示すように、ユーザ指定ファイル34とバックアップファイル33はそれぞれ1個であるが、ユーザ指定ファイル34とバックアップファイル33は、ユーザ用途に応じて、それぞれ複数個存在する構成でも良い。

【0143】

【発明の効果】本発明によれば、ユーザがバックアップ取得を希望するファイル（指定ファイル）についてのみバックアップを取得するので、少ないハードディスク容量でのファイル2重化が可能であり、安価なハードウェアの設備投資でファイル2重化が実現できる。

【0144】また、既存のプログラムに改造を加えずに回復を実行する手順をユーザが選択でき、例えば、回復実行の要求を他のプログラムから通知することにより回復の実行に他のプログラムが介在することができ、ユーザニーズに応じた複数の回復手順を選択できる。

【図面の簡単な説明】

【図1】ファイル2重化システムの一実施例を示すブロック図である。

【図2】本実施例における管理テーブル内のバックアップ区分および回復区分の構成を示す図である。

【図3】バックアップ区分毎の複写動作の違いを表す説明図である。

【図4】本実施例における起動停止処理部の主処理動作例を示すフローチャートである。

【図5】本実施例における起動停止処理部による入力処理動作例を示すフローチャートである。

【図6】本実施例における管理テーブルの構成例を示す説明図である。

【図7】本実施例におけるテーブル参照処理部による処理動作例を示すフローチャートである。

【図8】本実施例における複写制御処理部の処理動作例を示すフローチャートである。

【図9】本実施例における複写処理部による処理動作例を示すフローチャートである。

【図10】本実施例における複写制御処理部の書込監視処理動作例を示すフローチャートである。

【図11】本実施例における複写処理部の実行と対に実行される時間監視処理動作例を示すフローチャートである。

【図12】本実施例における要求通知部の処理動作例を示すフローチャートである。

【図13】本実施例におけるA P Pと要求通知部との間で受け渡される情報の例を示す図である。

【図14】本実施例における複写制御処理部で起動される更新終了監視処理動作例を示すフローチャートである。

【図15】本実施例における複写制御処理部で起動される通知監視処理動作例を示すフローチャートである。

【図16】本実施例における要求通知部による処理動作例を示すフローチャートである。

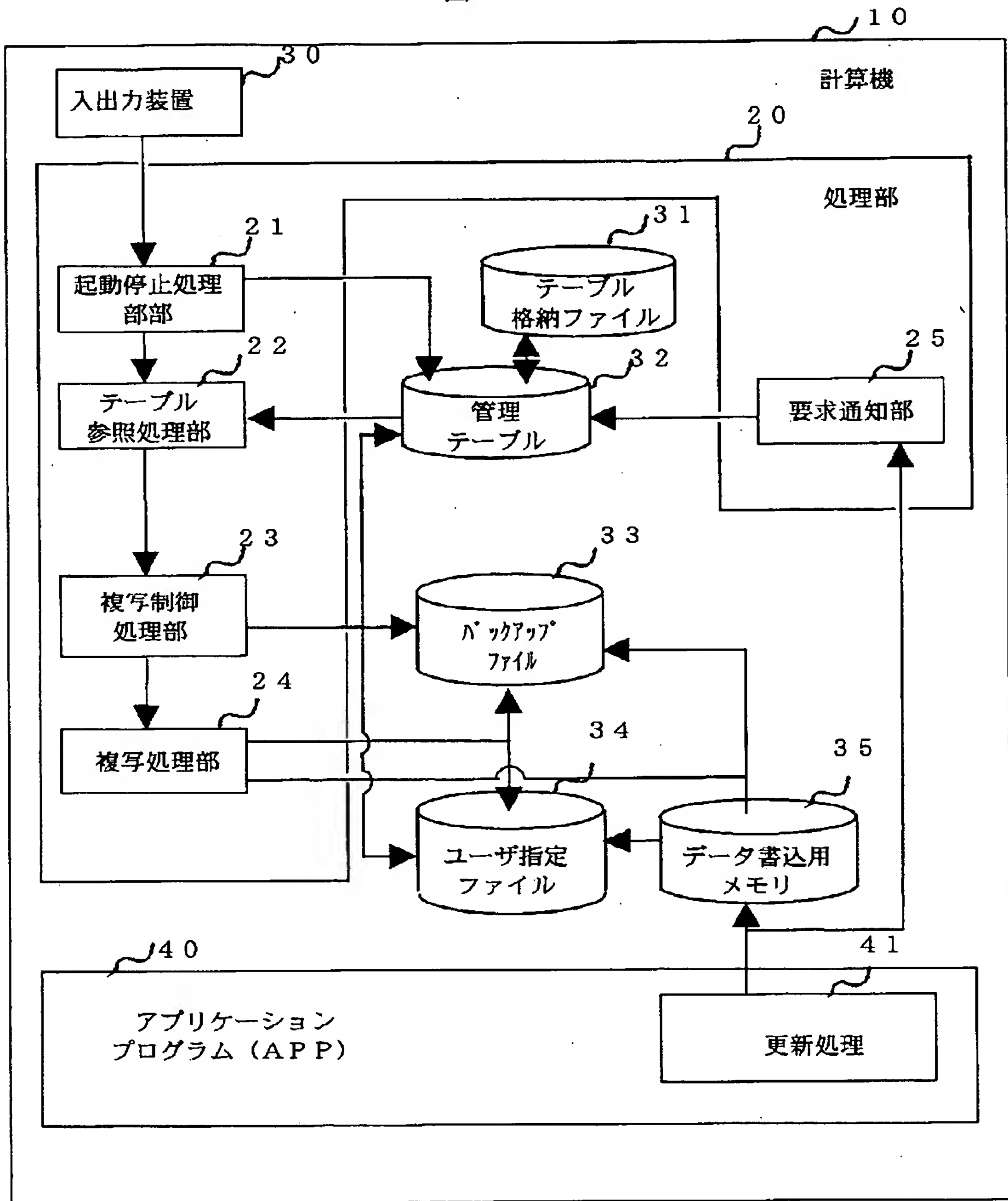
【図17】本実施例における複写制御処理部で起動されるエラー監視処理動作例を示すフローチャートである。

【符号の説明】

10…計算機、20…処理部、21…起動停止処理部、22…テーブル参照処理部、23…複写制御処理部、24…複写処理部、25…要求通知部、30…入出力装置、31…テーブル格納ファイル、32…管理テーブル、33…バックアップファイル、34…ユーザ指定ファイル、35…データ書込用メモリ、40…A P P、41…更新処理。

【図1】

図1



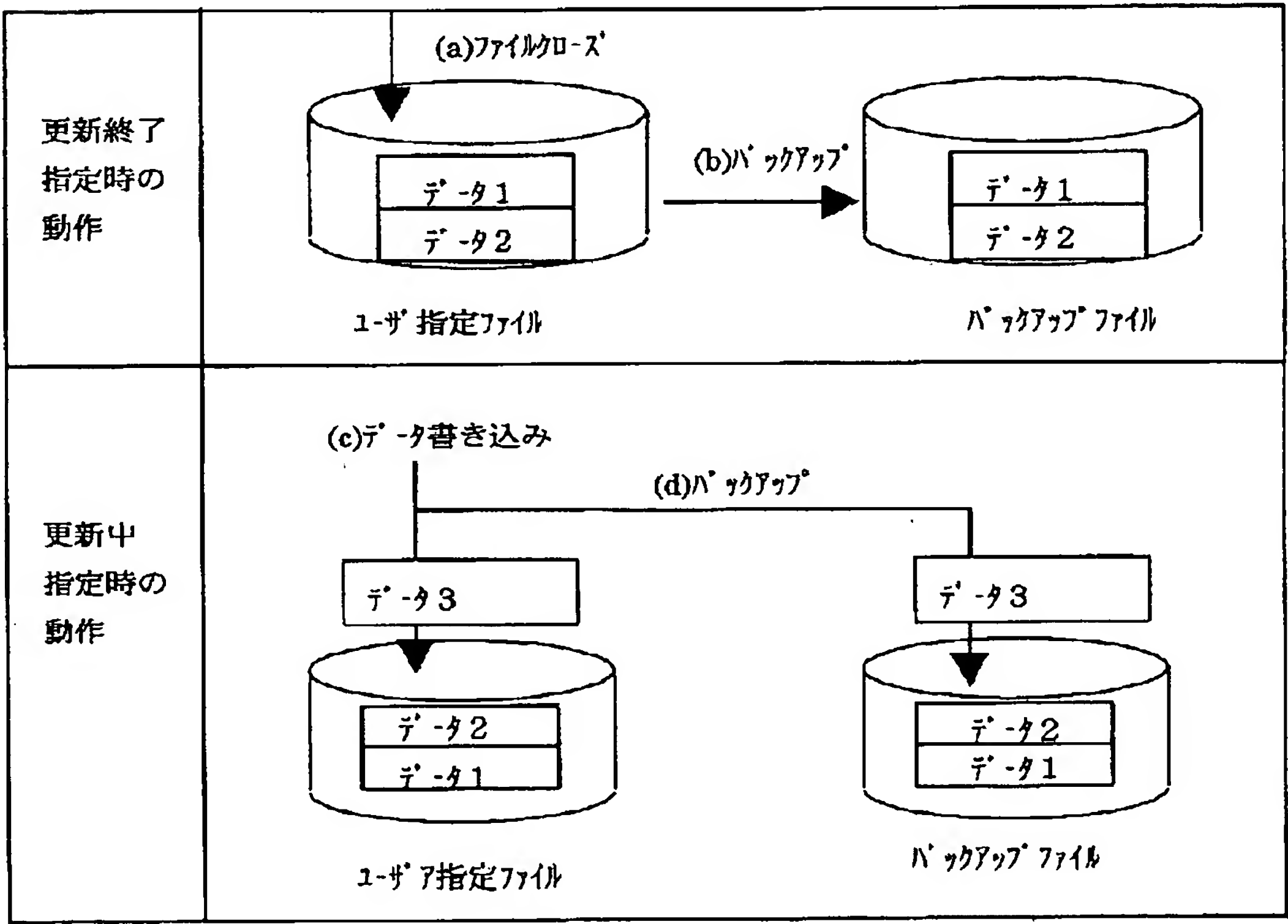
【図2】

図2

機 能	実行区分	実行区分に対応するシステム動作内容
バックアップ	更新終了	ユーザ 指定ファイルが更新完了された時点でバックアップ 実行
	更新中	ユーザ 指定ファイルの更新オープン中にデータ書き込みが発生した時点でバックアップ 実行
回復	自動	ユーザ 指定ファイルにデータ入出力エラーが発生した時点で回復実行
	手動	ユーザ 任意の実行契機で、ユーザ 手動操作により回復実行
	通知	他プログラムからの通知を受けた時点で本システムが回復実行

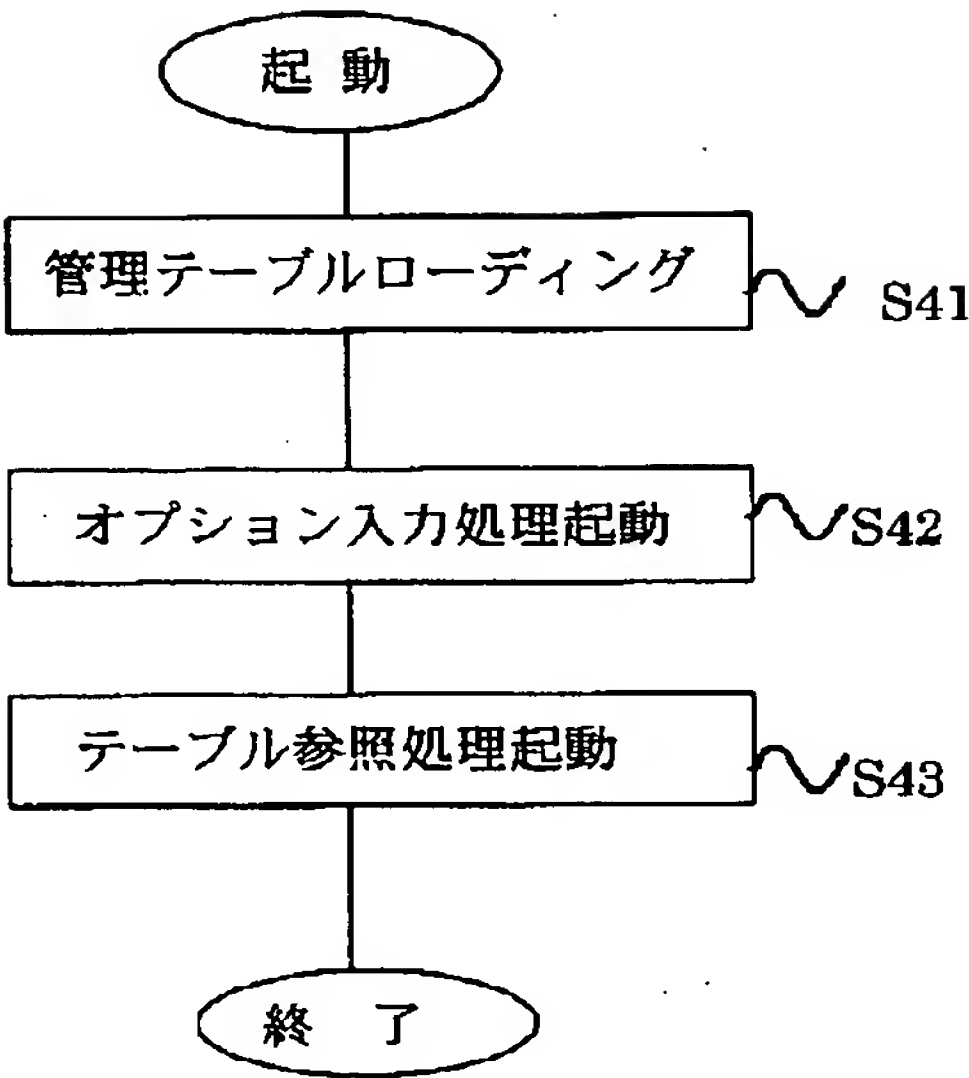
【図3】

図3



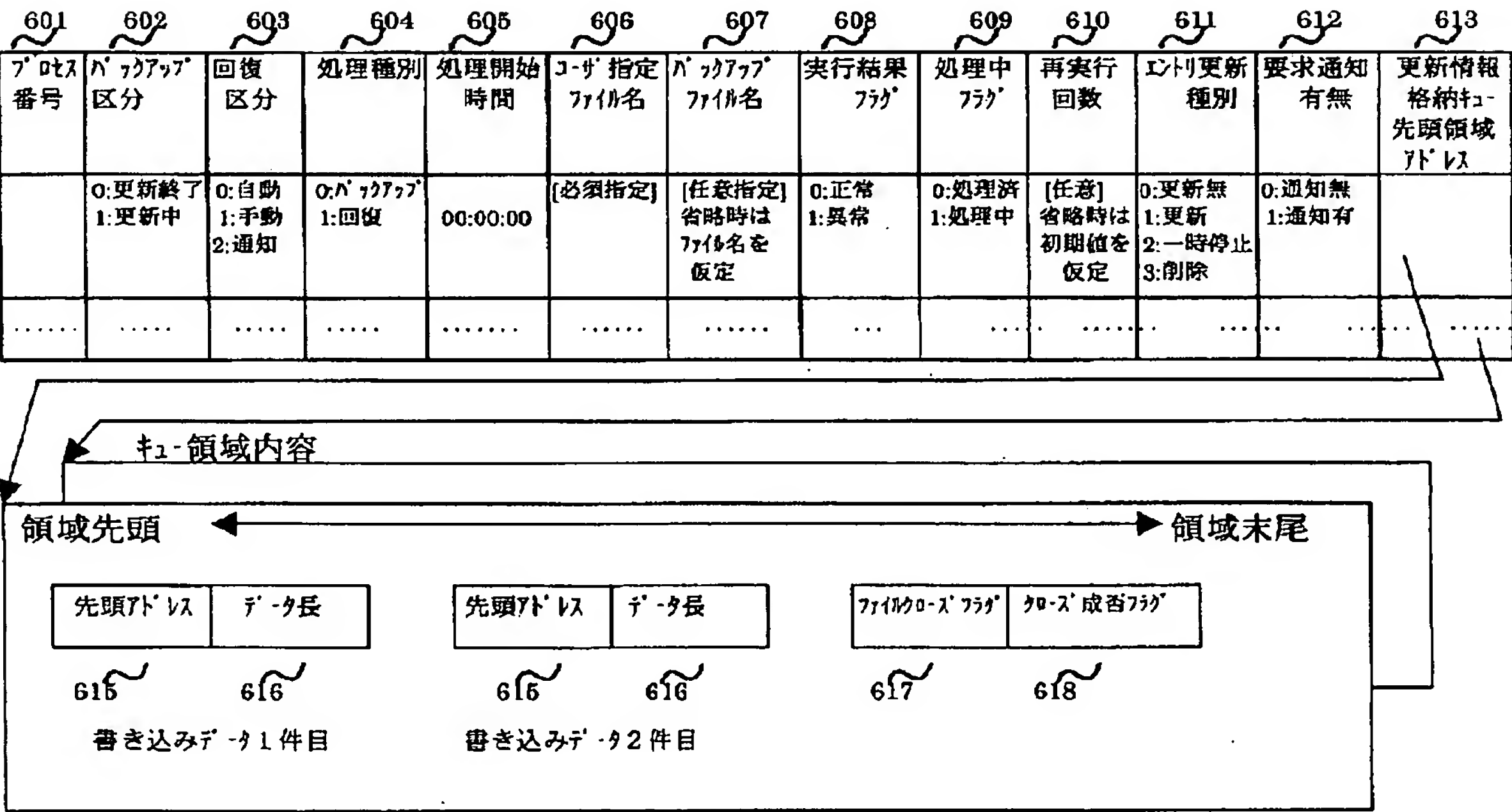
【図4】

図4



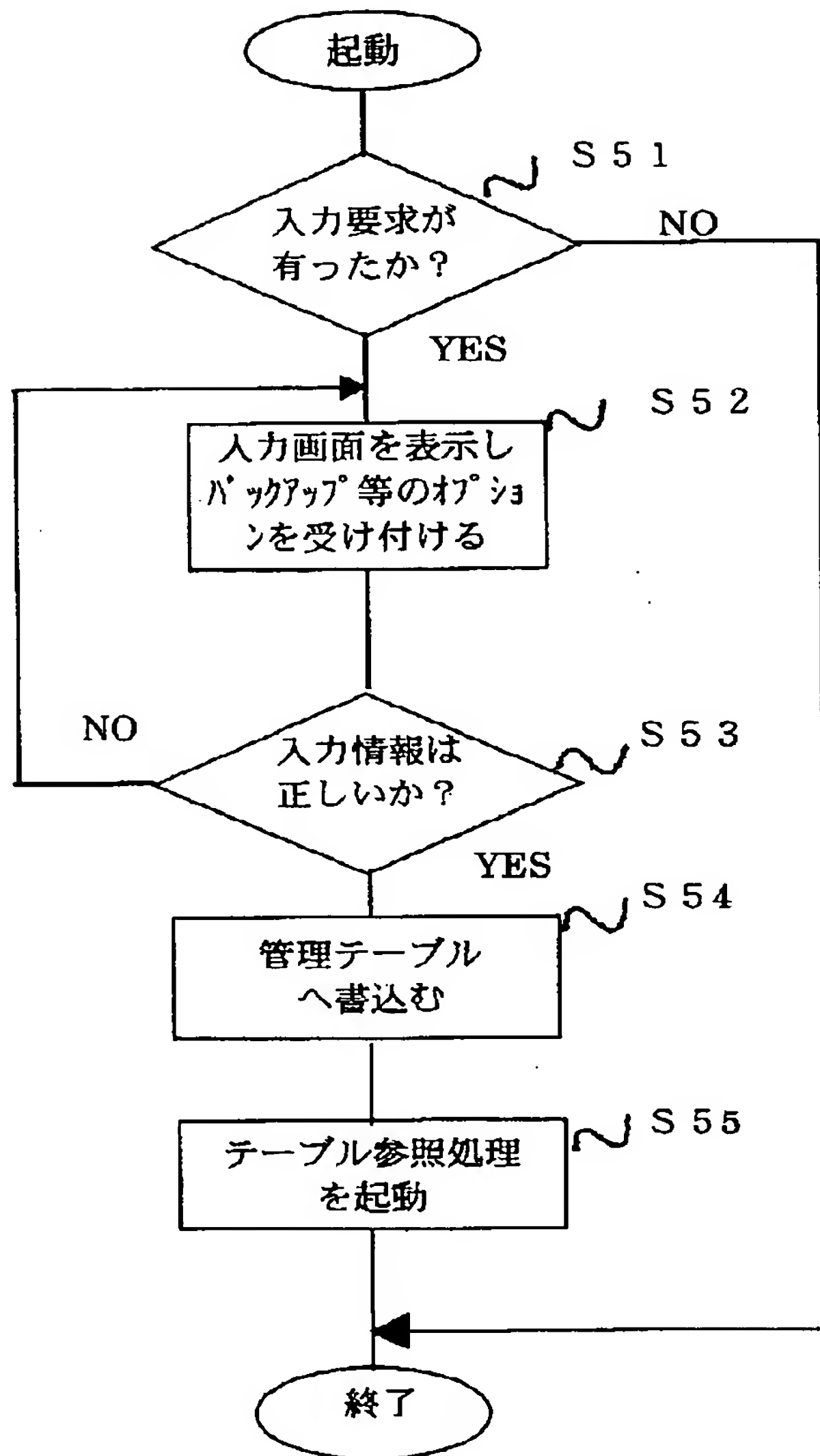
【図6】

図6



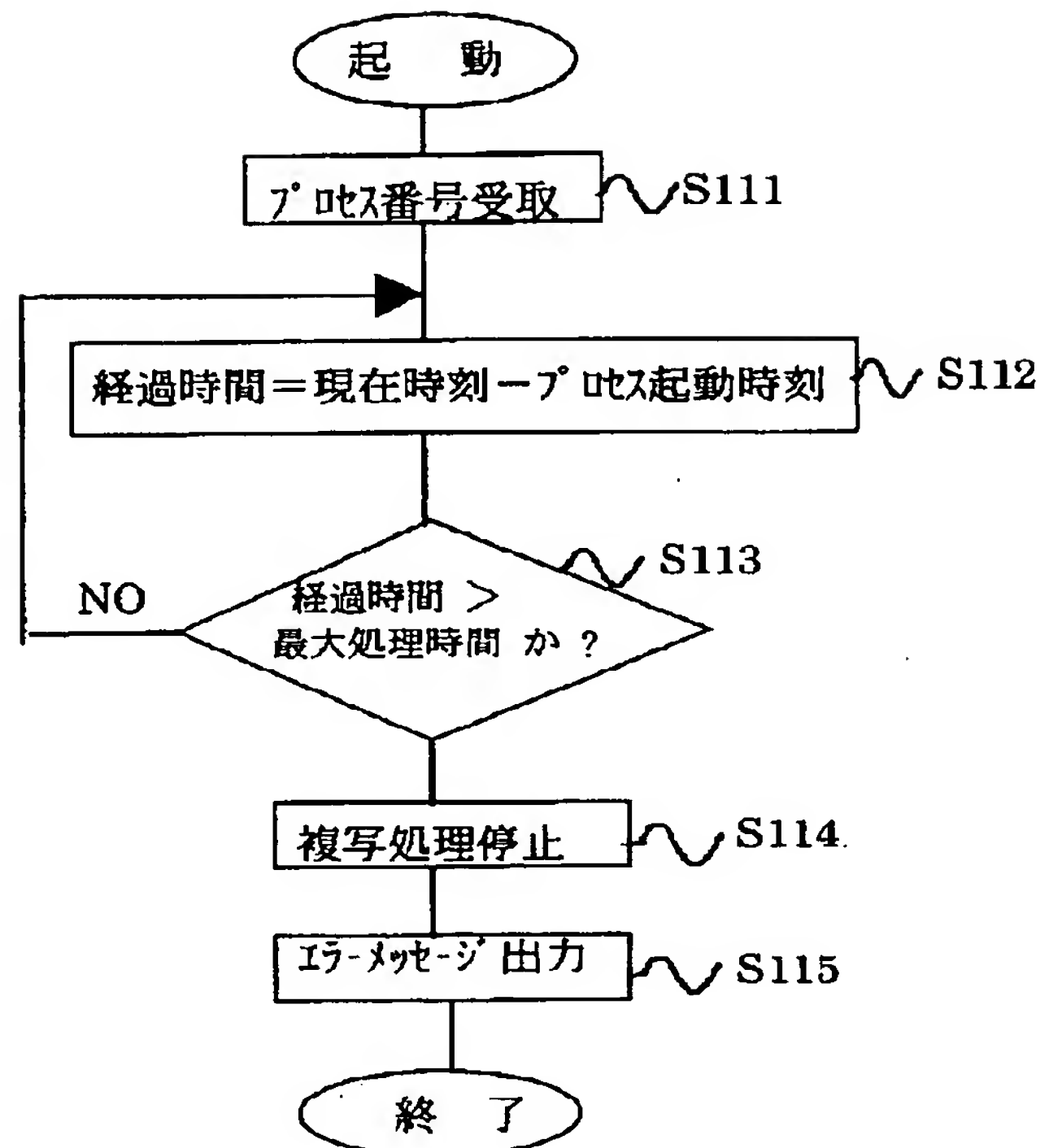
【図5】

図5



【図11】

図11

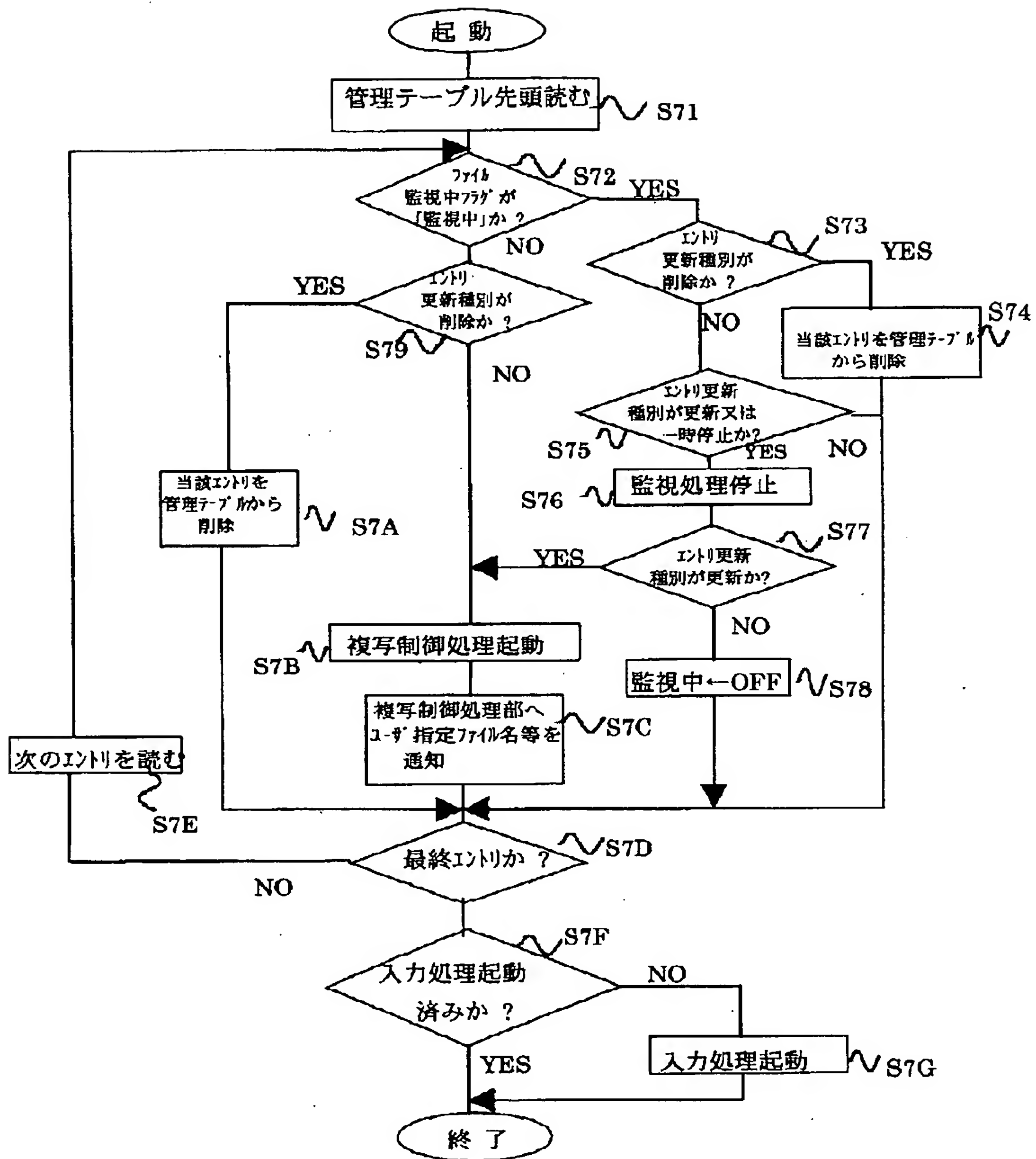


【図13】

図13

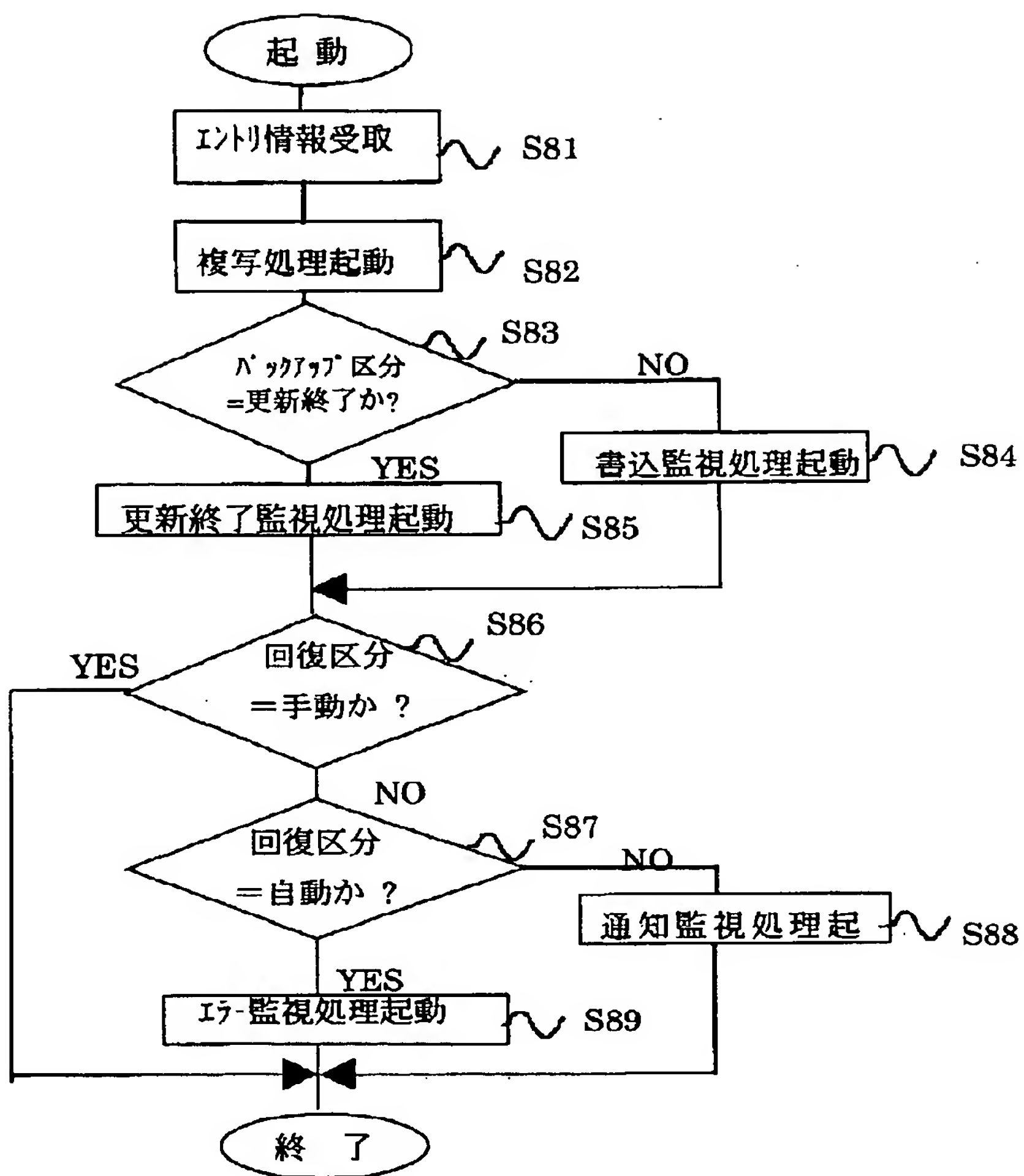
バックアップ/回復種別	第一引数	第二引数	第三引数	第四引数	返却値
バックアップ	ユーザ指定 ファイル名	更新データ 先頭アドレス	更新 データ長	ファイルクロス 結果	通知 結果
回復	ユーザ指定 ファイル名	—	—	—	通知 結果

Figure 7 shows a schematic diagram of a rectangular structure. It consists of a central vertical line segment and two horizontal line segments, one above and one below the vertical segment. The vertical segment is labeled with the letter 'a' at its top end, and the horizontal segments are labeled with the letter 'b' at their right ends. The entire structure is enclosed within a rectangular frame.



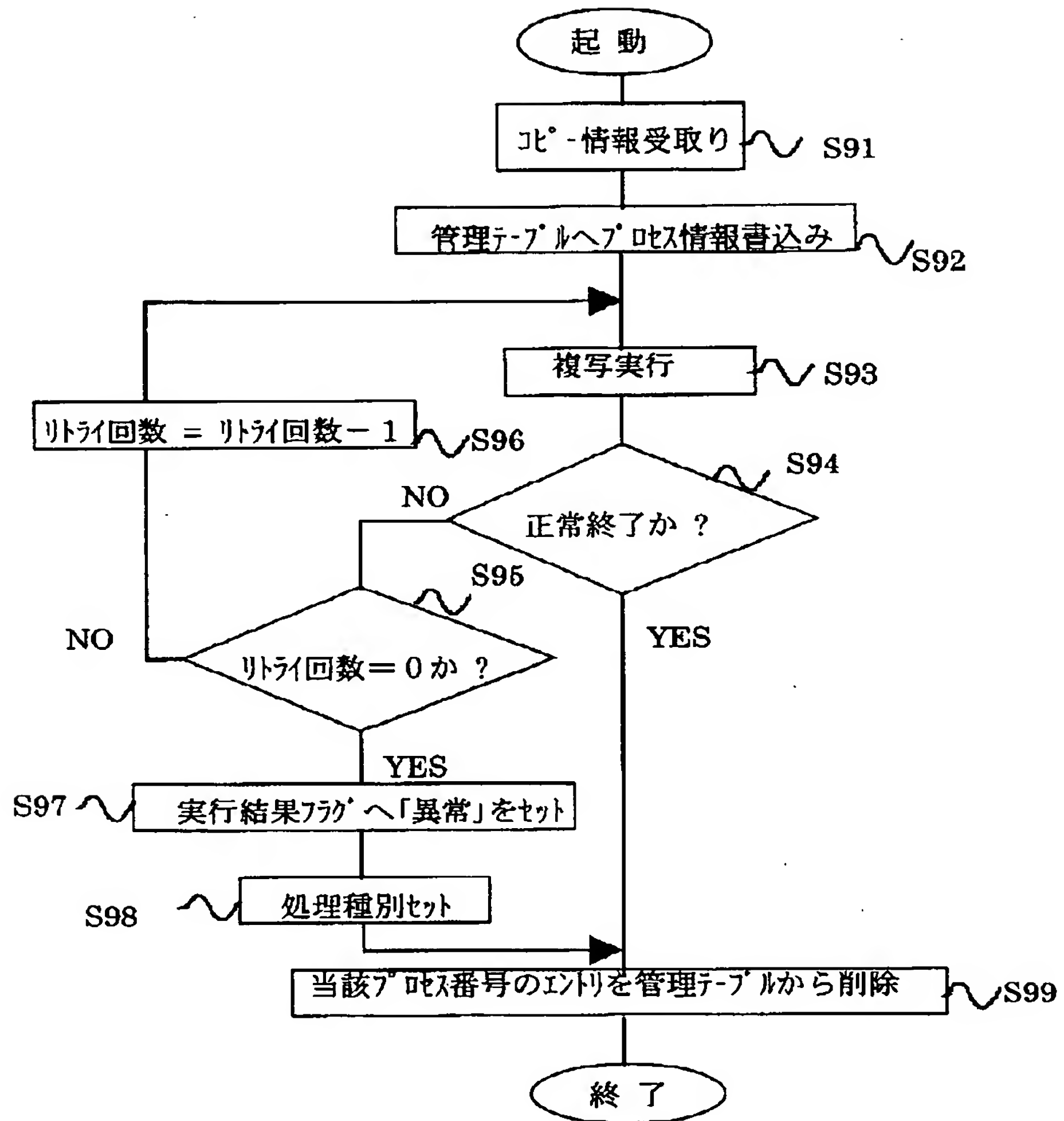
【図8】

図8



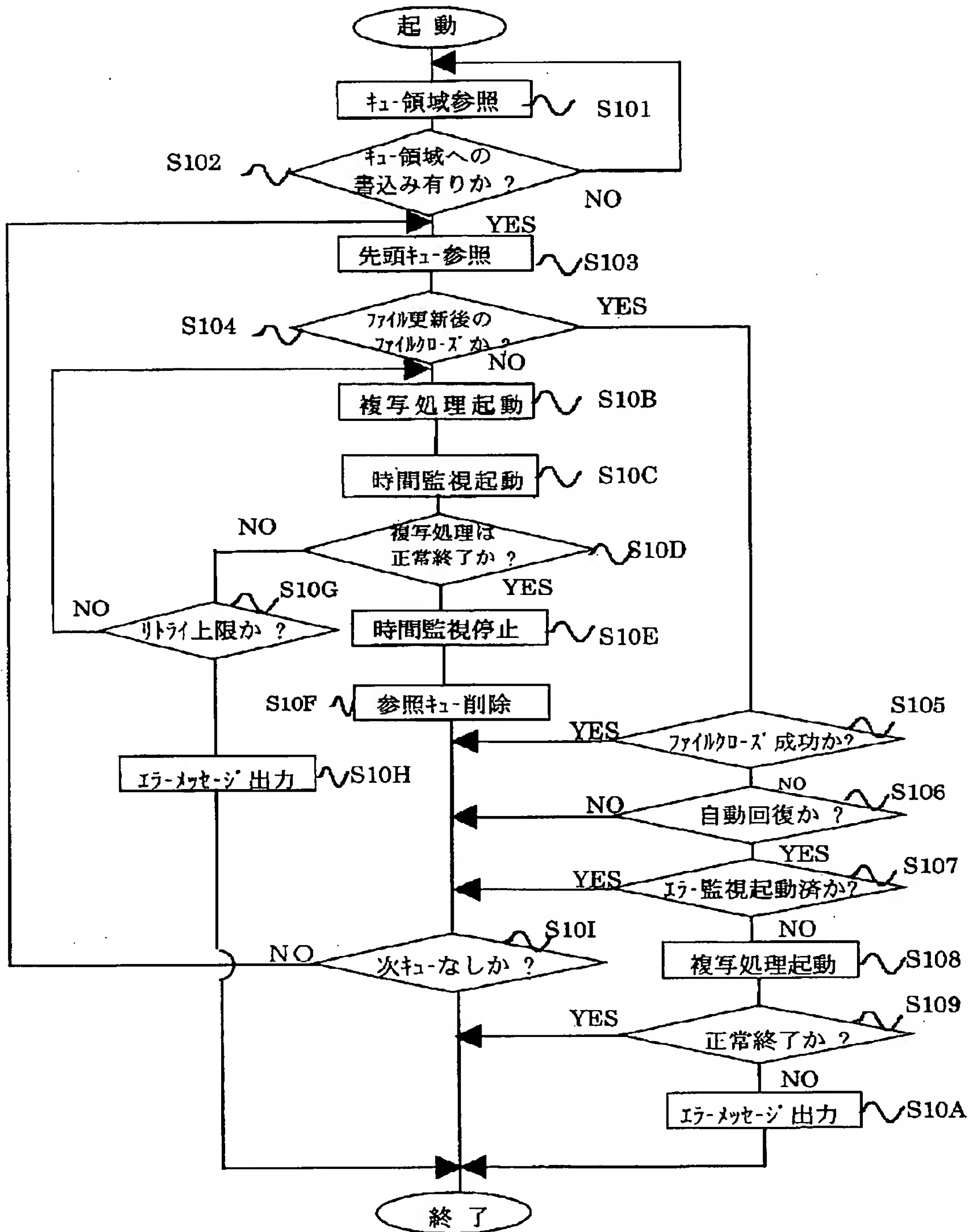
【図9】

図9



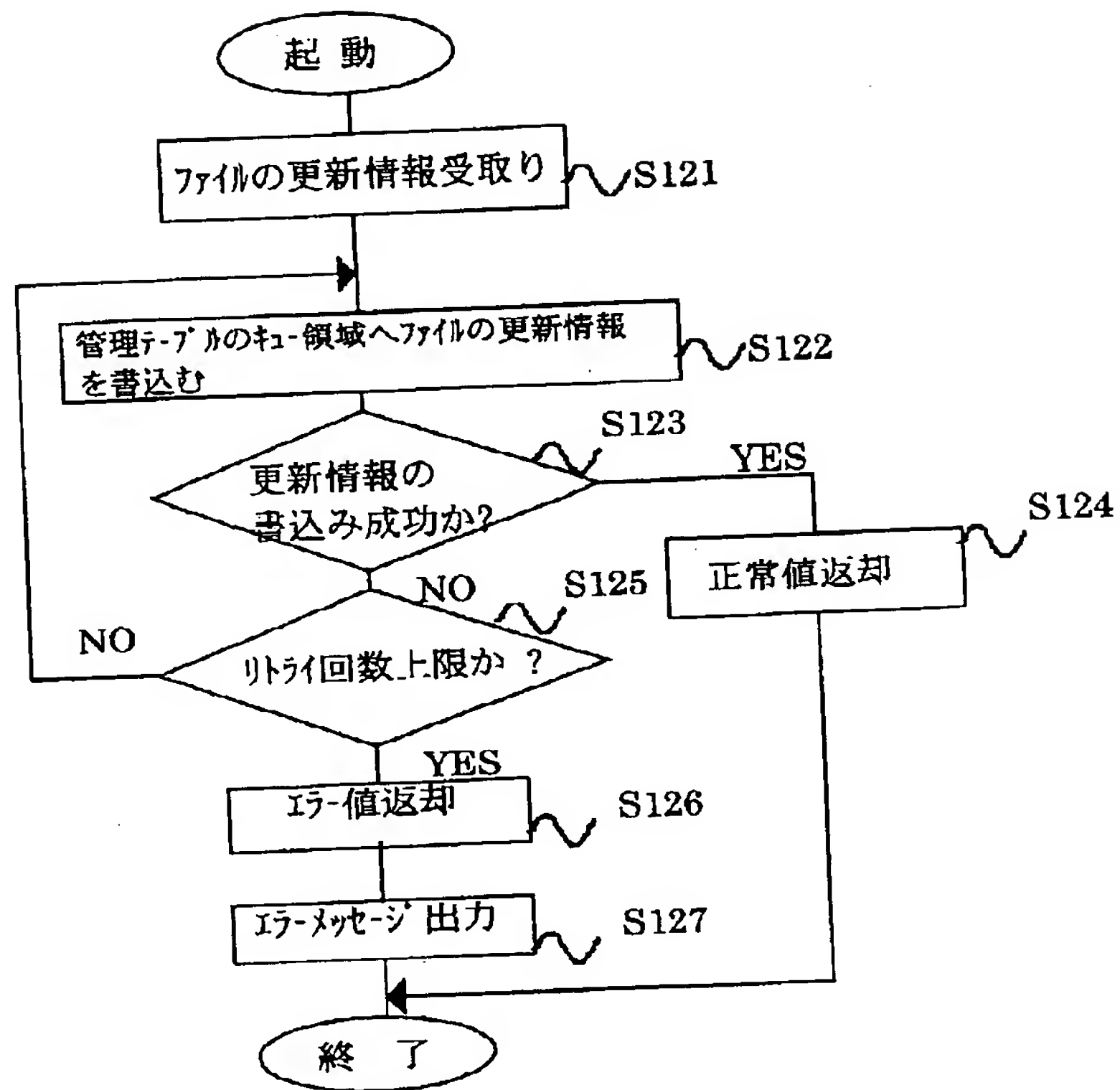
【図10】

図10



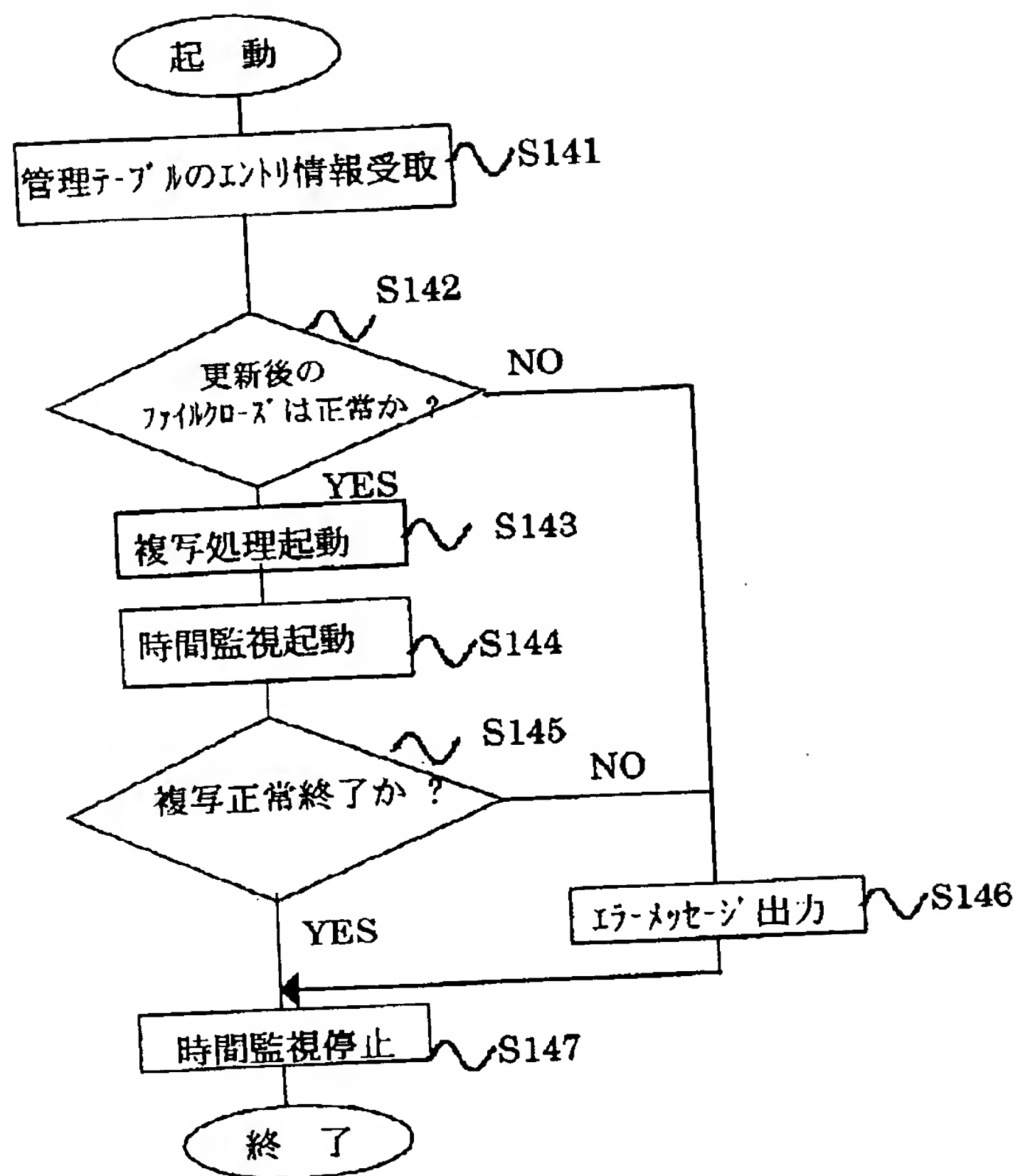
【図12】

図12



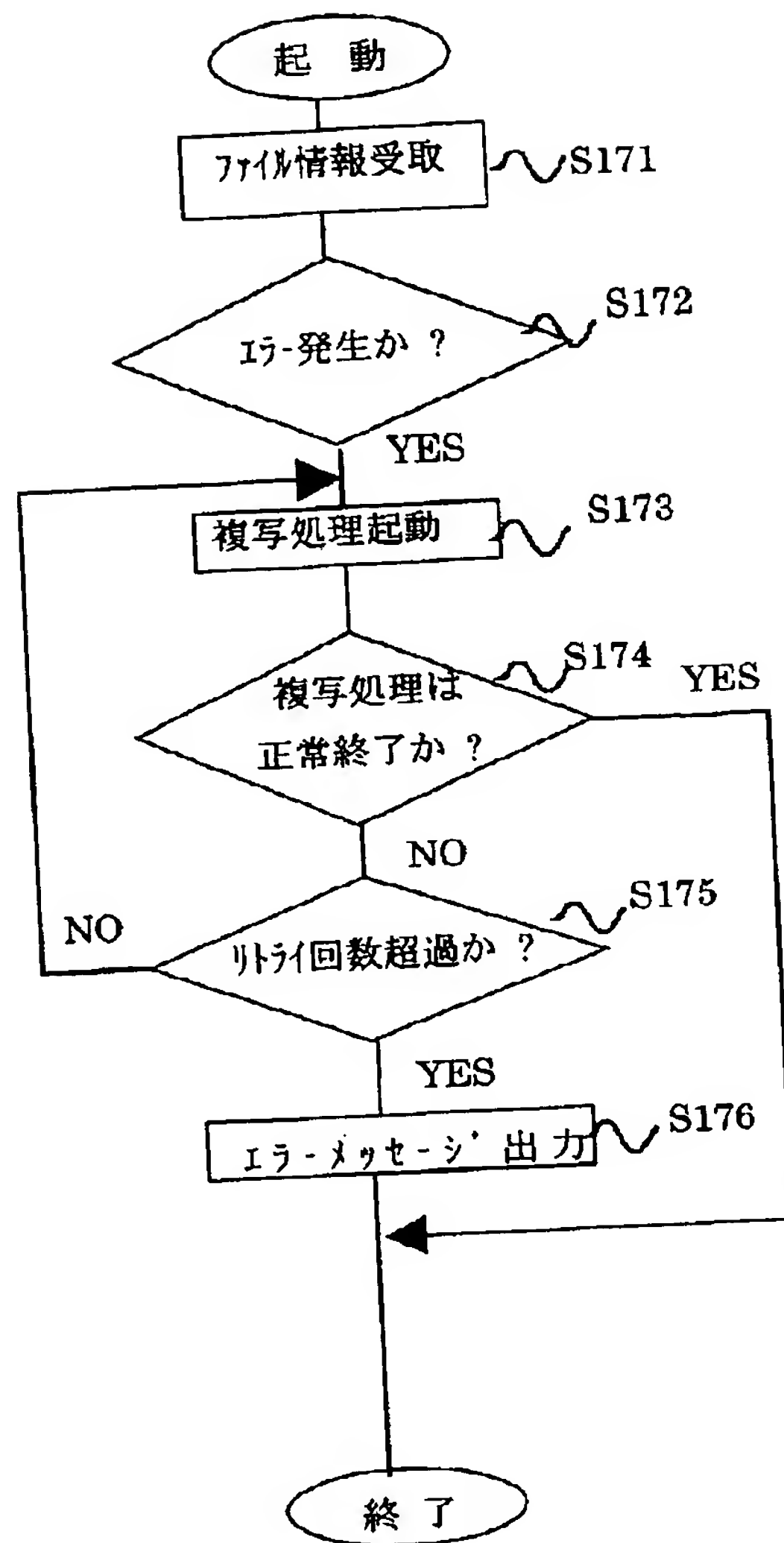
【図14】

図14



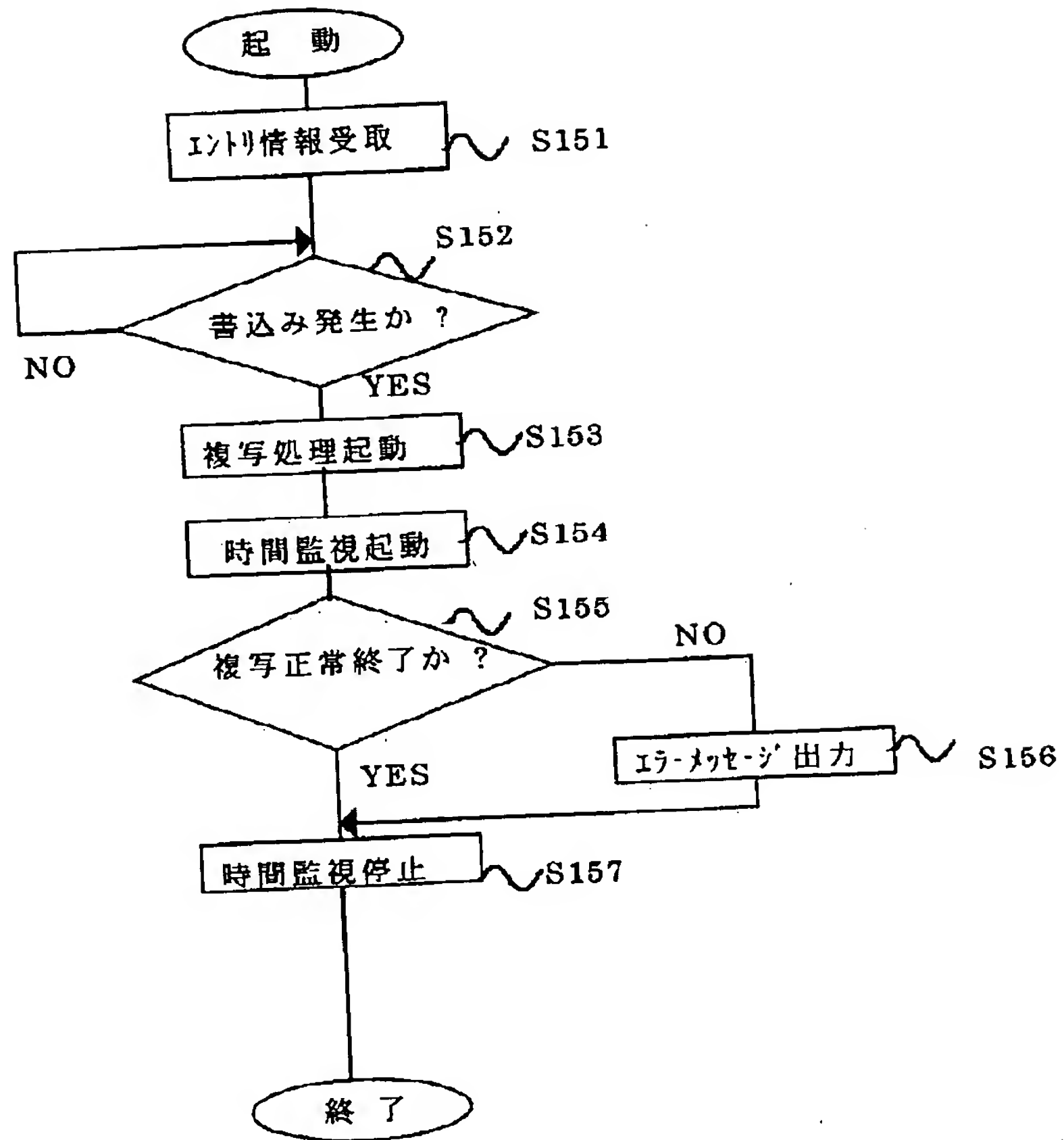
【図17】

図17



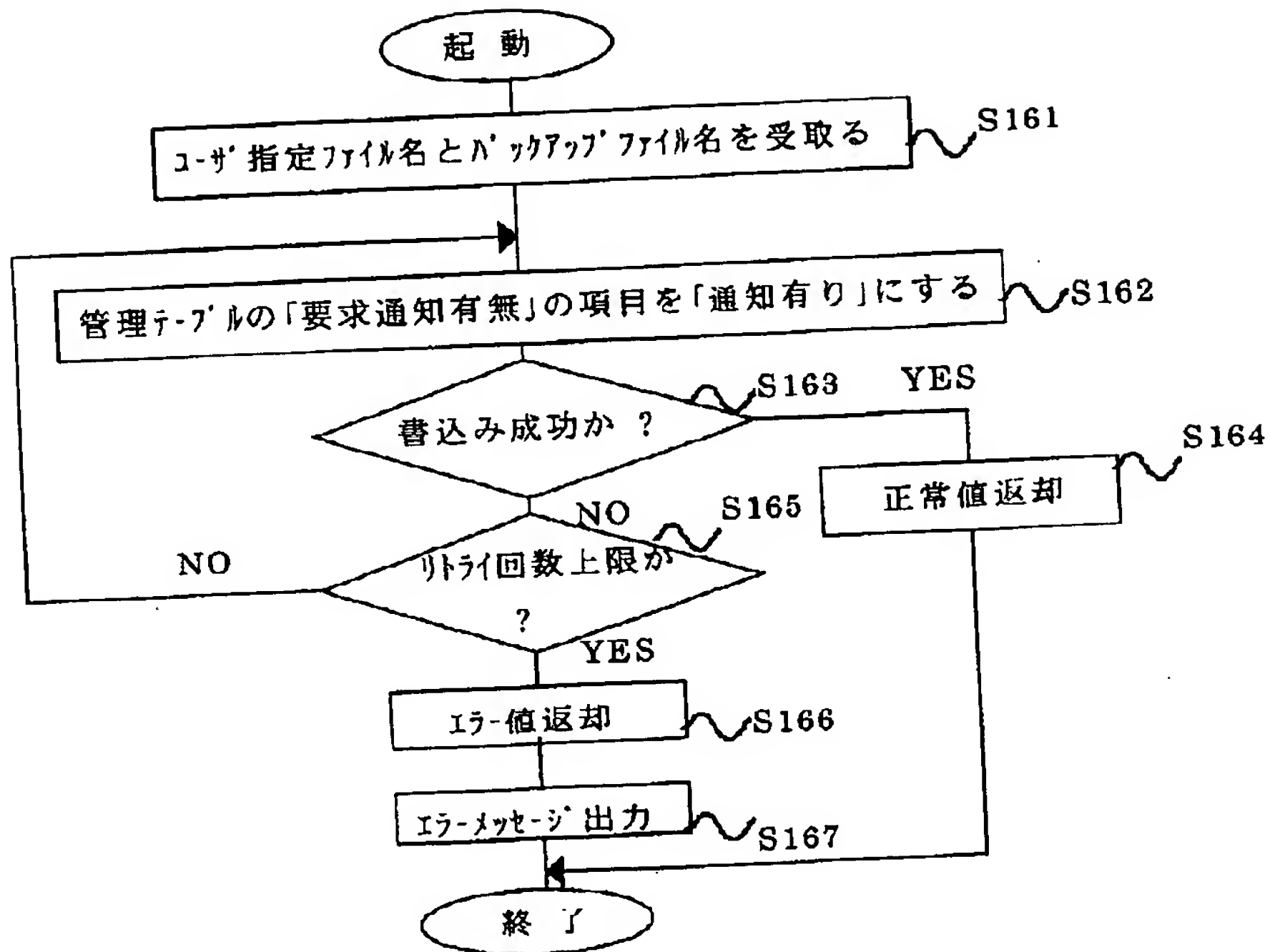
【図15】

図15



【図16】

図16



フロントページの続き

(72)発明者 森本 隆宏
神奈川県横浜市戸塚区戸塚町5030番地 株
式会社日立製作所ソフトウェア事業部内

(72)発明者 川勝 大輔
神奈川県横浜市戸塚区戸塚町5030番地 株
式会社日立製作所ソフトウェア事業部内